Efficiency Improvement Using Two Balanced Subsets

HongTae Kim*

요 약

Efficiency is one of the most important factors in cryptographic systems. Cheon et al. proposed a new exponent form for speeding up the exponentiation operation in discrete logarithm based cryptosystems. It is called split exponent with the form $e_1+\alpha e_2$ for a fixed element α and two elements e_1, e_2 with low Hamming weight representations. They chose e_1, e_2 in two unbalanced subsets S_1, S_2 of Z_p, respectively. We achieve efficiency improvement making S_1, S_2 balanced subsets of Z_p. As a result, speedup for exponentiations on binary fields is 9.1% and speedup for scalar multiplications on Koblitz Curves is 12.1%.

두 개의 balanced subset을 이용한 효율성 개선

김 홍 태*

ABSTRACT

암호시스템에서 효율성은 매우 중요한 요소 중의 하나이다. 천정희 외 3인은 이산대수 문제에 기반하는 암호 시스템에서 지수승 연산 속도를 높이기 위해 새로운 지수 형태를 제안하였다. 제안된 변형은 고정된 원소 α와 작은 해밍 웨이트를 가지는 두 원소 e_1, e_2에 대해 $e_1+\alpha e_2$로 표현되며 스플릿 지수라 불린다. 그들은 e_1, e_2를 각각 Z_p의 부분집합이면서 언밸런스드 부분집합인 S_1, S_2에서 선택하였다. 본 논문에서는 S_1, S_2를 Z_p의 부분집합이면서 밸런스드 부분집합이 되도록 하여 효율성을 개선한다. 결과적으로, 이진 유한체에서의 지수승 연산 속도는 9.1%, 코블리츠 곡선에서의 스칼라 곱셈 연산 속도는 12.1% 빨라진다.

Key words : Exponentiation, Split exponent, Low Hamming weight, Discrete logarithm

접수일(2018년 2월 18일), 게재확정일(2018년 3월 23일)

* 공군사관학교/기초과학과

1. Introduction

Discrete logarithm(DL) is a very important concept in cryptography. Many cryptographic systems are based on this. In one direction, studies on these are related to protocols or attacks. In other direction, there is about the efficiency of DL-based cryptosystems. To make these systems efficient, there have been many trials to reduce the complexity. The complexity is related to exponentiation operation in these systems. Exponentiation consists of two parts, squarings and multiplications. The number of squarings is fixed and squarings can be reduced by precomputations or usage of special structure likewise normal basis. Meanwhile, the number of multiplications in standard exponent has been the focus of many studies [1, 4, 6, 7]. In addition to these trials, variants of standard exponent have been presented [5, 2]. Hoffstein and Silvermann proposed a system with special exponent of products of random small Hamming weights [5]. Cheon et al. introduced an exponent with special type $e_1+\alpha e_2$ for a fixed element α where e_1, e_2 have low Hamming weights [2]. The exponentiation operation using exponent of theirs is faster than exponentiation algorithms for standard exponents. We improve the efficiency of Cheon et al.'s system using two balanced subsets of Z_p.

Result We can get efficient result using two balanced subsets. The number of multiplications for binary fields is reduced to 30 from 33 by 9.1% compared to Cheon et al.'s method. A scalar multiplication on Koblitz curves requires 29 point additions, which is a speedup by 12.1% compared to Cheon et al.'s method.

Organization The remainder of this paper is organized as follows. In Section 2, we introduce Cheon et al.'s split exponent DL problem and mention relevant properties. Also, we give two algorithms for two balanced subsets. In Section 3, we analyze the performance of exponentiations on binary fields and multiplications on Koblitz curves. We conclude in Section 4.

2. Split Exponent DL Problem for good α's

Split exponent is an element of $S_1+\alpha S_2$ of Definition 2.1. The definition of split exponent discrete logarithm(SEDL) problem is given in [2].

Definition 2.1 ([2]) Let S_1, $S_2 \subseteq Z_p$. Let P be a generator of an additive abelian group G with prime order p. Let α be an element in Z_p. Let A be any probabilistic algorithm. A's success probability in solving the SEDL problem on (S_1, S_2, α, G) is defined by

$$Adv^{SEDL}_{A,S_1,S_2,\alpha,G} := \Pr[A(P,xP) = x \mid x \leftarrow^r S_1+\alpha S_2]. \quad (1)$$

Definition 2.2 Let S_1, $S_2 \subseteq Z_p$. We call $\alpha \in Z_p$ "c-good" for S_1, S_2 if $S_1+\alpha S_2 \geq cp$.

Theorem 2.1 Let S_1, $S_2 \subseteq Z_p$. If α is c-good for S_1, S_2 then the SEDL problem on (S_1, S_2, α, G) is at least $(t, \frac{\epsilon}{c})$-hard if the DL problem on G is (t, ϵ)-hard.

Proof. If $|S_1+\alpha S_2| \geq cp$ then a random

$x \in Z_p$ is in $S_1 + \alpha S_2$ with probability c. Therefore if algorithm A breaks SEDL on (S_1, S_2, α, G) in time t with probability $\frac{\epsilon}{c}$ then A breaks DL with probability ϵ. □

We revise Cheon et al.'s algorithms to pick c-good α's for a good enough constant c. We can reduce the complexity using balanced subsets S_1, S_2 for randomized algorithm.

Algorithm 1 Choosing a Random Split Exponent

On input S_1, S_2, α :

1. Randomly select $z \leftarrow^r Z_p$.
2. For i=1,2, 3, ⋯:
   2.1 Randomly select $y \leftarrow^r S_2$
   2.2 Check if $x = z - y\alpha \in S_1$. Output (x,y) and stop if it is.
3. If no $(x,y) \in S_1 \times S_2$ is found, go to State 1.

Algorithm 2 Finding a c-good Element α

Let $\tilde{c} = 2c$.

On input S_1, $S_2 \subseteq Z_p$ and $\tau \in Z$:

1. Randomly select $\alpha \leftarrow^r Z_p$.
2. Randomly select $x_1, x_2, ..., x_\tau \in Z_p$ and test if each of them belongs to $S_1 + \alpha S_2$ by Step 2 of Algorithm 1.
3. If the number of x_i's in $S_1 + \alpha S_2$ is at least $\tilde{c}\tau$, output α. Otherwise go to State 1.

Algorithm 1 outputs a uniformly distributed element of $S_1 + \alpha S_2$ with the expected number of $\frac{1}{c}$ iterations provided that α is c-good. Hence the expected running time of this algorithm is at most $O(\frac{\sqrt{|S|}}{c})$ modular multiplications where $|S| = \max\{|S_1|, |S_2|\}$.

Let $X_1, X_2, ..., X_\tau$ be random variables such that $X_i = 1$ if $x_i \in S_1 + \alpha S_2$ and 0 otherwise. Let $X = X_1 + ... + X_\tau$. Using this notation, $P_{fail} = \Pr[X > \tilde{c}\tau]$ for $\tilde{c} = 2c$. By a Chernoff bound, Cheon et al. get the bound as follows:

$$P_{fail} < 2^{-0.557c\tau}. \quad (2)$$

We can estimate the expected running time of Algorithm 2. Let $\theta = \frac{|S_1||S_2|}{p} > 1$. A random element $\alpha \in Z_p$ is $\tilde{c}$-good with probability

$$P_1 \geq \frac{\theta/(\theta+1) - \tilde{c}}{1 - \tilde{c}}. \quad (3)$$

If α is $\tilde{c}$-good, the probability that at least $\tilde{c}\tau$ elements from randomly selected τ elements from Z_p belong to $S_1 + \alpha S_2$ is given by

$$P_2 = \sum_{i \geq \tilde{c}\tau} \binom{\tau}{i} \tilde{c}^i (1-\tilde{c})^{\tau - i} \approx \frac{1}{2}. \quad (4)$$

If $|S_1||S_2| \geq p$ and we use Algorithm 1 to test if x_i's belongs to $S_1 + \alpha S_2$, the expected running time of Algorithm 2 is at most

$$T = (P_1 P_2)^{-1} \times \frac{\tau}{2c} \times |S|. \quad (5)$$

The following example gives the bound for specific parameters. This bound is more than that of Cheon et al.'s.

Example If $\theta \geq 1$, we may take $c=\frac{1}{4}$ and $\tau=576$ as Cheon et al.'s example. Then $P_{fail} \leq 2^{-80}$, $P_1 \geq \frac{\theta-1}{\theta+1}$ and $P_2 \approx \frac{1}{2}$. Thus, the running time of Algorithm 2 is $T=2304\frac{\theta+1}{\theta-1}|S|$, which is less than $2^{13}|S|$ for $\theta \geq 2$.

3. Application of Split Exponents

In this section, we improve the efficiency of Cheon et al.'s method using two balanced subsets.

3.1 w-NAF Representation

The definition of w-NAF is given in [9, 3, 8]:

Definition 3.1 Let w be an integer ≥ 2 and D a subset of Z with $0 \notin D$. A w-NAF with the digit set D is a sequence of digits satisfying the following two conditions:

1. Each non-zero digit belongs to D.
2. Among any w consecutive digits, at most one is non-zero.

The number of w-NAFs of length $\leq m$ and weight t with a coefficient set D is given as follows [3]:

$$\binom{m-(w-1)(t-1)}{t}|D|^t. \quad (6)$$

3.2 Split Exponent for Binary Fields

Let x be a split exponent $x=x_1+\alpha x_2$ for unsigned w-NAFs x_1 and x_2 with length m where t_1 and t_2 are weights of x_1 and x_2, respectively. We can compute g^x in two ways where g is a finite field element. The first method is to individually compute g^{x_1} and $(g^{\alpha})^{x_2}$ and multiply them. The number of multiplications is $t_1+t_2+2^w-3$. Cheon et al. [2] reduced the number of multiplications in the Algorithm 3 using the BGMW technique [1]. Its complexity is given by $t_1+t_2+2^{w-1}-2$. We denote $x_j=\sum_{i=0}^{m-1} x_{j,i}2^i$, where $x_{j,i} \in D=\{1,3,5,...,2^w-1\}$ for $j=1,2$. If two sets S_1, S_2 of $S_1+\alpha S_2 \subset Z_p$ are chosen with balanced size, we can get the merit in efficiency.

Algorithm 3 Exponentiation by a split exponent

1. Input g,h,x_1 and x_2.
2. Set $s \leftarrow 1$, $t \leftarrow 1$.
3. For $i=2^w-1,2^w-3,...,3,1$:
   3.1 For each j such that $x_{1,j}=i$, set $s \leftarrow s \times g^{2^j}$.
   3.2 For each j such that $x_{1,j}=i$, set $s \leftarrow s \times h^{2^j}$.
   3.3 If $i=1$ then set $t=t \times s$; else set $t=t \times s^2$.
4. Output t.

Table 1 compares Cheon et al.'s performance with that of ours when m is the largest integer less than or equal to $\lfloor \log p \rfloor - w+1$. To guarantee the 2^{80} security, Cheon et al. assumed that the exponentiation algorithms use 160-bit exponents and $|S_1||S_2| \approx 2^{160}$. They used unbalanced sets and different weights. According to Table 1, the expected speedup is 9.1% for $|S_1|=|S_2| \approx 2^{81.4}$ where $w=2$, $m=159$. Cheon et al. used $|S_1|=1.2 \times 2^{120}$, $|S_2|=1.0 \times 2^{40}$ and different weights $t_1=27$, $t_2=6$.

<Table 1> Performance of exponentiations for 160-bit exponent (β: the number of multiplications)

		t_1	t_2	$\|S_1\|$	$\|S_2\|$	β
$w=2$	Ours	15	15	$2^{81.4}$	$2^{81.4}$	30
$m=159$	Cheon	27	6	$2^{120.2}$	$2^{40.0}$	33
$w=3$	Ours	13	12	$2^{84.5}$	$2^{79.5}$	27
$m=158$	Cheon	22	5	$2^{121.7}$	$2^{39.1}$	29
$w=4$	Ours	11	10	$2^{84.0}$	$2^{77.9}$	27
$m=157$	Cheon	19	4	$2^{124.6}$	$2^{36.1}$	29
$w=5$	Ours	10	9	$2^{86.7}$	$2^{79.7}$	33
$m=156$	Cheon	16	4	$2^{123.1}$	$2^{40.0}$	34
$w=6$	Ours	9	8	$2^{87.7}$	$2^{79.6}$	47
$m=155$	Cheon	13	4	$2^{116.6}$	$2^{43.7}$	47

3.3 Split Exponent for Scalar Multiplications on Koblitz Curves

A τ-adic w-NAF is a w-NAF as a τ-adic representation of an integer with the nonzero digit set $D=\{\pm 1,\pm 3,...,\pm(2^w-1)\}$. It is denoted by $a=\sum_{i=0}^{m-1} a_i\tau^i$. Let E be an ordinary elliptic curve. Given a τ-adic w-NAF $a=\sum_{i=0}^{m-1} a_i\tau^i$ and a point $Q\in E$, aQ is defined as $aQ=\sum_{i=0}^{m-1} a_i\tau^i(Q)$.

Let k be the form $k=k_1+\alpha k_2$ and $Q=\alpha P$ for $P,Q\in E$. Cheon et al. reduced the number of point additions by sharing the point additions in the precomputation stage. They compute

$$kP=k_1P+k_2Q=R_1+3R_3+...+(2^{w-1}-1)R_{2^{w-1}-1}$$

using the BGMW technique [1] where

$$R_i=k_{1,j=\pm i}sign(k_{1,j})\tau^j(P)+k_{2,j=\pm i}sign(k_{2,j})\tau^j(Q)$$

for $k_1=\sum_{j=0}^{m-1} k_{1,j}\tau^j$ and $k_2=\sum_{j=0}^{m-1} k_{2,j}\tau^j$. The detailed procedure is given in Algorithm 4. It requires $wt(k_1)+wt(k_2)+2^{w-2}-2$ point additions, one point doubling, and $2(m-1)\tau$ operations as Cheon et al.'s results.

Algorithm 4 Scalar multiplication by a split scalar

1. Input P, Q, k_1 and k_2.
2. Precomputation stage:
   2.1 Set $R_i \leftarrow O$ for $i=1,3,5,...,2^{w-1}-1$.
   2.2 For $j=0$ upto $m-1$:
      2.2.1 Set $R_{|k_{1,j}|}\leftarrow R_{|k_{1,j}|}+sign(k_{1,j})\tau^j(P)$.
      2.2.2 Set $R_{|k_{2,j}|}\leftarrow R_{|k_{2,j}|}+sign(k_{2,j})\tau^j(Q)$.
   2.3 For $j=0$ upto $m-1$:
      2.3.1 For $i=1,3,5,...,2^{w-1}-1$:
         2.3.1.1 If $|k_{1,j}|=i$ or $|k_{2,j}|=i$,
            $R_i\leftarrow R_{|k_{1,j}|}+R_{|k_{2,j}|}$.
3. Computation of k_1P+k_2Q
   $=R_1+3R_3+...+(2^{w-1}-1)R_{2^{w-1}-1}$:
   3.1 Set $S\leftarrow R_{2^w-1}$; Set $T\leftarrow R_{2^w-1}$.
   3.2 For $i=2^{w-1}-3, 2^{w-1}-5,...,5,3$:
      3.2.1 Set $S\leftarrow S+R_i$.
      3.2.2 Set $T\leftarrow T+S$.
   3.3 Set $T\leftarrow 2T$.
   3.4 Set $T\leftarrow T+S+R_1$.
4. Output T.

Table 2 shows results for specific parameters. Cheon et al. used different $|S_1|,|S_2|,t_1$ and t_2 from ours for same w and m. If we use a normal basis, we can ignore the computation of τ map. In this case, our method is faster than Cheon et al.'s method by 12.1% for $w=2, m=157$. Cheon et al. used $|S_1|=1.1\times 2^{122}$, $|S_2|=1.8\times 2^{39}$ and different weights $t_1=28, t_2=6$.

<Table 2> Performance of scalar multiplications on

K163 (A: Addition, D: Doubling, T:Computation of the τ map, γ: the number of scalar multiplications)

		t_1	t_2	$\lvert S_1\rvert$	$\lvert S_2\rvert$	γ
$w=2$	Ours	15	15	$2^{81.1}$	$2^{81.1}$	$29A+312T$
$m=159$	Cheon	27	6	$2^{120.2}$	$2^{40.0}$	$33A+312T$
$w=3$	Ours	13	12	$2^{84.5}$	$2^{79.5}$	$25A+1D+310T$
$m=158$	Cheon	22	5	$2^{121.7}$	$2^{39.1}$	$28A+1D+310T$
$w=4$	Ours	11	10	$2^{84.0}$	$2^{77.9}$	$23A+1D+306T$
$m=157$	Cheon	19	4	$2^{124.6}$	$2^{36.1}$	$25A+1D+306T$
$w=5$	Ours	10	9	$2^{86.7}$	$2^{79.7}$	$25A+1D+302T$
$m=156$	Cheon	16	4	$2^{123.1}$	$2^{40.0}$	$27A+1D+302T$
$w=6$	Ours	9	8	$2^{87.7}$	$2^{79.6}$	$31A+1D+298T$
$m=155$	Cheon	13	4	$2^{116.6}$	$2^{43.7}$	$32A+1D+298T$

4. Conclusion

Studies on efficiency of cryptographic systems have been proposed sustainedly. Recently, variants of original exponent have been come into the spotlight. Though strict security proof is not perfect, many trials have been suggested. We only have focused on the efficiency of split exponents. It would be interesting to apply the previous security proof techniques to schemes or protocols using split exponents.

참고문헌

[1] E. Brickell, D. Gordon, K. McCurley and D. Wilson, "Fast exponentiation with precomputation", Proceedings of Eurocrypt 1992, LNCS, Vol. 658, Springer-Verlag, pp. 200-207. 1993.

[2] J. H. Cheon, S. Jarecki, T. Kwon and M. Lee, "Fast Exponentiation Using Split Exponents", IEEE Transactions on Information Theory, Vol. 57, No. 3, pp. 1816-1826, 2011.

[3] J. H. Cheon and J. H. Yi, "Fast batch verification of multiple signatures", Proceedings of Public Key Cryptography 2007, LNCS, Vol. 4450, Springer-Verlag, pp. 442-457, 2007.

[4] P. D. Rooij, "Efficient exponentiation using precomputation and vector addition chain", Proceedings of Eurocrypt 1994, LNCS Vol. 950, Springer-Verlag, pp. 389-399, 1994.

[5] J. Hoffstein and J. H. Silverman, "Random small hamming weight products with applications to cryptography", Discrete Applied Mathematics, Vol. 130, No. 1, pp. 37-49, 2003.

[6] D. Knuth, 'The art of computer programming', volume 2: seminumerical algorithms, Addison-Wesley, 3rd edition, 1998.

[7] C. H. Lim and P. J. Lee, "More flexible exponentiation with precomputation", Proceedings of Crypto 1994, LNCS, Vol. 839, Springer-Verlag, pp. 95-107, 1994.

[8] J. Muir and D. Stinson, "Minimality and other properties of the width-w non-adjacent form", Mathematics of Computation, Vol. 75, pp. 369-384, 2006.

[9] J. Solinas, "Efficient arithmetic on Koblitz curves", Designs, Codes and Cryptography, Vol. 19, pp. 195-249, 2000.

〔저자소개〕

김 홍 태 (HongTae Kim)
2003년 2월 서울대 수리과학부 학사
2006년 2월 서울대 수리과학부 석사
2013년 2월 서울대 수리과학부 박사
2013년 2월 ~ 현재 공군사관학교 기초과학과 수학교수
email : yeskafa@naver.com